

AMENDMENTS TO THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A method of communicating data, via a device driver, between an application and an interface having a feature to which an interface identifier is assigned, ~~the assignment of the interface identifier to the feature being susceptible to change after an event,~~ the method comprising:
 - storing a logical identifier corresponding to the feature;
 - providing the logical identifier to the application for directing communication associated with the feature between the device driver and the application; and
 - maintaining correspondence between the logical identifier and the feature independently of the interface identifier assigned to the feature so that communication between the application and the device driver directed using the logical identifier remains associated with the feature following a change in the assignment of the interface identifier to the feature after an event,wherein the application, the interface, and the device driver communicate internal to a device.
2. (Currently Amended) A method according to Claim 1, wherein communication between the interface and the device driver is directed based on the interface identifier.
3. (Previously Presented) A method according to Claim 1, including compiling a list of logical identifiers and corresponding interface identifiers for the feature if the feature meets a predetermined criterion.
4. (Previously Presented) A method according to Claim 1, wherein the device driver is arranged to communicate the interface identifier assigned to the logical identifier to the application on request.
5. (Previously Presented) A method according to Claim 1, wherein the device driver is arranged to accept requests from the application to define connections between physical devices to a bus using the logical identifier in place of the interface identifier.

6. (Previously Presented) A method according to Claim 1 wherein the application is arranged to communicate with the device driver via device manager means.
7. (Previously Presented) A method according to Claim 1 wherein the feature of the interface comprises a peripheral connected to the interface and the interface identifier comprises a physical address assigned to the peripheral, the logical identifier comprising a logical address assigned to the peripheral.
8. (Previously Presented) A method according to Claim 7, wherein said maintaining correspondence includes interrogating the peripheral to which the logical address is assigned to determine the physical address assigned to the peripheral following a bus reset.
9. (Currently Amended) A method according to Claim 7, wherein the ~~driver~~ device driver is arranged to communicate the interface identifier assigned to the logical identifier to the application on request, and further comprising communicating the interface identifier for the peripheral by communicating the physical address of the peripheral and communicating a unique node identifier containing further information identifying the peripheral.
10. (Previously Presented) A method according to Claim 1, wherein the feature of the interface comprises a channel of defined parameters available via the interface and the interface identifier comprises an interface channel number, the logical identifier comprising a logical channel identifier.
11. (Previously Presented) A method according to Claim 10, wherein the device driver is arranged to receive a request from the application to allocate the channel of defined parameters and to return the logical channel identifier if allocation is successful.
12. (Previously Presented) A method according to Claim 10, wherein the device driver is arranged to accept a preferred interface channel number and to allocate a preferred interface channel if available, and to allocate a free channel if the preferred interface channel is not available or if the preferred interface channel is not specified.
13. (Previously Presented) A method according to Claim 10, wherein the device driver is arranged to receive an identifier of a preferred interface channel, to recognize a pre-determined

key in place of a valid interface channel number as indicating that the preferred interface channel is not specified, and to report an error to the application if other invalid interface channel numbers are specified.

14. (Previously Presented) A method according to Claim 10, wherein the device driver is arranged to communicate the interface channel number to the application, and at least one other parameter selected from: a maximum rate allocated to the channel; a rate currently available; a number of connections using the channel; and identifiers of each connection using the channel.

15. (Previously Presented) A method according to Claim 1 wherein the device driver is arranged to accept requests from the application to define one or more connections between physical devices attached to the interface by reference to logical addresses and logical channel identifiers.

16. (Previously Presented) A method according to Claim 1 wherein the device driver is arranged to establish at least a broadcast connection.

17. (Previously Presented) A method according to Claim 1 wherein the device driver is arranged to signal the event to the application, the event including reset of a bus or a change in a bus topology or a change in a channel or a change in connection parameters.

18. (Currently Amended) A device driver for effecting communication between an the application and an interface having a feature to which an interface identifier is assigned, ~~the interface identifier being liable to change after an event~~, the device driver comprising:

means for storing a logical identifier corresponding to an interface identifier;

means for providing the logical identifier to the application for directing communication associated with the feature between the device driver and the application; and

means for maintaining correspondence between the logical identifier and the feature independently of the interface identifier assigned to the feature so that communication between the application and the device driver directed using the logical identifier remains associated with the feature following a change in the assignment of the interface identifier to the feature after an event,

wherein the application, the interface, and the device driver communicate internal to a device.

19. (Previously Presented) A device driver according to Claim 18, wherein the device driver is implemented in software.
20. (Previously Presented) A device driver according to Claim 18, wherein the device driver is arranged to compile a list of logical identifiers and corresponding interface identifiers for the feature if the feature meets a predetermined criterion.
21. (Previously Presented) A device driver according to Claim 18 including means for communicating the interface identifier assigned to the logical identifier to the application on request.
22. (Currently Amended) A device driver according to Claim 18, including means for accepting a request from the application to define connections between physical devices connected to a bus using the logical identifier in place of the interface identifier.
23. (Previously Presented) A device driver according to Claim 18, wherein the feature of the interface comprises a peripheral connected to the interface and the interface identifier comprises a physical address assigned to the peripheral, the logical identifier comprising a logical address assigned to the peripheral.
24. (Previously Presented) A device driver according to Claim 23, arranged to interrogate the peripheral to which the logical address is assigned to determine the physical address assigned to the peripheral following a bus reset.
25. (Previously Presented) A device driver according to Claim 23, including means for communicating the interface identifier assigned to the logical identifier to the application on request, and further comprising means for communicating the interface identifier for the peripheral by communicating the physical address of the peripheral and means for communicating a unique node identifier containing further information identifying the peripheral.
26. (Previously Presented) A device driver according to Claim 18, wherein the feature of the interface comprises a channel of defined parameters available via the interface and the interface identifier comprises an interface channel number, the logical identifier comprising a logical channel identifier.

27. (Currently Amended) A device driver according to Claim 26 including channel allocating means arranged to receive a request from the application to allocate the channel of defined parameters and to return the logical channel identifier if allocation is successful.
28. (Previously Presented) A device driver according to Claim 27, wherein the channel allocating means is arranged to accept a preferred interface channel number and to allocate a preferred interface channel if available, and to allocate a free channel if the preferred interface channel is not available or if the preferred interface channel is not specified.
29. (Previously Presented) A device driver according to Claim 27, wherein the channel allocating means is arranged to receive an identifier of a preferred interface channel, to recognize a pre-determined key in place of a valid interface channel number as indicating that the preferred interface channel is not specified, and to report an error to the application if other invalid interface channel numbers are specified.
30. (Previously Presented) A device driver according to Claim 26, including means for communicating the interface channel number to the application, and at least one other parameter selected from: a maximum rate allocated to the channel; a rate currently available; a number of connections using the channel; and identifiers of each connection using the channel.
31. (Previously Presented) A device driver according to Claim 18 including means arranged to accept requests from the application to define one or more connections between physical devices attached to the interface by reference to logical channel identifiers.
32. (Previously Presented) A device driver according to Claim 18, including means arranged to establish at least a broadcast connection on request by the application.
33. (Previously Presented) A device driver according to Claim 18, including means for signaling the event to the application, the event including reset of a bus and a change in a bus topology or a change in a channel or a change in connection parameters.
34. (Canceled)
35. (Previously Presented) A data processing system according to Claim 47 implemented in a receiver/decoder which includes means for receiving broadcast data, the interface means being

arranged for connection to a digital video recorder or a digital display device or a computer for display or storage of at least a portion of the received data.

36. (Previously Presented) A receiver/decoder according to Claim 35, wherein the device driver means is arranged to cooperate with further device driver means for modifying the broadcast data to produce a modified data stream for passing to said interface means.

37. (Previously Presented) A receiver/decoder according to Claim 35, wherein the interface means conforms to an IEEE 1394 standard or a variant thereof.

38. (Previously Presented) A receiver/decoder according to Claim 35, wherein the application is run in an interpreted language and the device driver means is compiled.

39. (Previously Presented) A receiver/decoder according to Claim 35, wherein the device driver means is arranged to transmit commands for controlling the digital video recorder from the application and/or to receive data concerning information stored on the digital video recorder.

40. (Previously Presented) A receiver/decoder according to Claim 39, wherein the data is in a MPEG format.

41. (Canceled)

42. (Canceled)

43. (Canceled)

44. (Canceled)

45. (Canceled)

46. (Canceled)

47. (Currently Amended) A data processing system comprising:

run-time engine means for running an application;

interface means for connection to a device, the interface means having a feature to which an interface identifier is assigned, ~~the interface identifier being liable to change after an event;~~ and

device driver means for effecting communication between the application and the interface means, the device driver means comprising:

means for storing a logical identifier corresponding to an interface identifier,

means for providing the logical identifier to the application for directing communication associated with the feature between the device driver means and the application, and

means for maintaining correspondence between the logical identifier and the feature independently of the interface identifier assigned to the feature so that communication between the application and the device driver means directed using the logical identifier remains associated with the feature following a change in an assignment of the interface to the feature after an event,

wherein the application, the interface, and the device driver communicate internal to a device.